Coding to get into MAANG (Meta, Amazon, Apple, Netflix, Google)

Tharun Vambaravelli

Pursuing his master's in the stream of Computer science and Engineering at International Institute of Information Technology , Hyderabad. Email: tharunvambaravelli@gmail.com

Skills such as Problem-solving, expertise in solving algorithms and data structures, coding competition expertise, etc. are required for getting into MAANG.

Language and Fundamentals

Firstly choose a language and understand the basic fundamentals concepts such as variable declaration, basic syntax, data type and structures,flow control structures (conditions and loops), functional programming, recursion, object-oriented programming. Preferably, choose an object oriented language.

Time and Space Complexity

For a problem, there can be more than one solution. You need to come up with an optimal solution for a problem. If you come up with multiple solutions for a problem, how do you choose the optimal one? Here comes the concept of time and space complexity. It is really important to learn these concepts and given a solution you must be able to analyze the time and space complexity.

Data structures

Understanding data structures is integral to participating in competitive programming, as you will be faced with making decisions on what data structure to utilize to most efficiently solve the problem you are given. Some important Data structures are: Arrays, Stacks, Queues, Linked List, Binary Tree, Binary Search Tree, Graphs, Tries, Hash tables, Heap. Some of the advanced data structures such as : Disjoint sets, Fibonacci Heap, Segment tree, Fenwick tree, B tree, B + tree, Self balancing binary tree (RB, AVL).

Learning data structures alone is not sufficient but also trying to implement these data structures in your chosen language will be useful for better understanding. Try to understand and analyze the time and space complexities of the operations performed on each data structure. This will help you in choosing an appropriate data structure for solving a given coding problem.

Algorithms

Algorithms are used to find the best possible way of solving a problem. Why to reinvent a wheel ? Algorithms can make life easier for developers by arming them with a range of general problem solving techniques. Choosing the standard algorithm is equally important as choosing an appropriate data structure for solving a problem. Some of the standard algorithms are: Searching & Sorting ,Greedy algorithms ,Dynamic Programming, Pattern Searching, Recursive and Backtracking algorithms, Geometric Algorithms, Graph Algorithms, Bit Algorithms, Number theory algorithms. Try to understand and analyze the time and space complexities for each algorithm. Give an attempt to implement each algorithm you have learnt in the chosen language.

Debugging

One common mistake a beginner makes is not using the debuggers. Try to use the debugger to debug the code and find out errors where you have made the mistake.

Practice

Try to solve at least 200 questions on any of the platforms like LeetCode, HackerRank, GeeksForGeeks, CodeChef, CodeForces etc. Try to stick to one or two coding platforms in the beginning phase until you are confident enough in solving questions and get into the habit of coding. Stick to learn topic wise (data structure and algorithms) in the early days of coding. Try to pick a data structure,understand and implement it , and solve problems on this chosen data structure. In a race of solving the problems, don't forget to build actually needed thought processes. It is more of quality rather than quantity while solving the problems. It is more important to solve problems from different topics.

Try to be consistent in solving problems i.e. try to solve at least 2 problems per day. Don't directly jump into solving the hard problems first as it may affect your confidence. Build confidence by first solving the easy problems,then medium and then hard problems.

Giving Contests and Competitions

After you think that you have got confidence in solving the problems, try to give contests in leetcode/codechef/ codeforces... Don't get demotivated if you can't solve a problem in the contest. Once the contest is finished, go through the discussion forums and try to understand various approaches. Even if you are able to solve the problem,try to check the other approaches which may be useful for solving other problems. Once you have enough practice in coding and giving contests regularly, you are now able to participate in competitions. Compete in competitions like ACM - ICPC, Google Code Jam, Facebook Hacker Cup. Happy coding :-)

About the Author

Tharun Sai is currently pursuing his master's in the stream of Computer science and Engineering at International Institute of Information Technology , Hyderabad.He is interested in Algorithms, Advanced Data Structures, Data systems,Operating systems.